

CLAIMS

1. (currently amended) A method for modifying a database file organized in segments and stored on a storage medium of limited rewritability, the method ~~containing~~ comprising the steps of:
 - reserving, within the database file, at least one area of predetermined size and position dedicated to writing thereto data records of at least one type, respectively;
 - indicating within the database file, as a last written segment that segment within the area to which data records were last written;
 - writing a specific data record of a specific type that is to be written to the database, into the next available segment or segments after the last written segment within the area dedicated to the specific type;
 - ~~comprising a step of:~~
 - continuing, whenever during the writing the end of the area has been reached, the writing at the first available segment of the area.
2. (currently amended) The method of claim 1, used for modifying a data record of a specific type in the database file wherein the database file contains an area dedicated to the specific type, and the method ~~contains~~ additionally comprises the steps of:
 - reading, from the area, the data record;
 - modifying the read data record;
 - obtaining a first write address information indicating a segment within the area to which a data record of the specific type was last written;

- forwarding, as part of ensuring distributed write, the first write address information so that it indicates a next segment within the area which contains unused space;
 - writing the modified data record to segments starting at the segment as indicated by the first write address information.
3. (currently amended) The method of claim 2, wherein the data record is a payload data record, the specific type is a "payload" type, the area is a payload area, the database file additionally has a control area and wherein the method additionally ~~contains~~ comprises the steps of:
- in case that an address information about the payload data record is contained in a control block within the control area, reading, from the control area, the control block;
 - updating the address information in the control block to reflect the first write address;
 - obtaining a second write address information indicating the segment within the control area to which a control block was last written;
 - forwarding, as part of ensuring distributed write, the second write address information so that it indicates a next segment within the control area which contains unused space;
 - writing the updated control block to the segment as indicated by the second write address information.

4. (currently amended) The method of claim 1, used for deleting a payload data record from the database file , wherein the database file contains a control area, and the method ~~contains~~ additionally comprises the steps of:
- reading, from the control area, control blocks containing information associated to the payload data record to be deleted;
 - marking, in the read control blocks, the payload data record to be deleted as deleted, thereby obtaining a modified control block;
 - obtaining a write address information indicating the segment within the control area to which a control block was last written;
 - forwarding, as part of ensuring distributed write, the write address information so that it indicates a next segment within the control area which contains unused space;
 - writing the modified control block to the segment as indicated by the forwarded write address information.
5. (previously presented) The method of claim 1, wherein the size of the segments corresponds to an integer multiple of the size of sectors or ECC blocks as defined in a physical format on the storage medium.
6. (previously presented) The method of claim 5, wherein the segments are allocated on the storage medium to be sector or ECC block aligned.
7. (previously presented) The method of claim 1, wherein the indicating is realized by attaching to a data record to be written a version count value which is incremented and taken modulo a predefined upper bound upon

each writing, the version count getting written to the database file as part of the data record being written thereto.

8. (previously presented) The method of claim 1, wherein the size of the at least one area is chosen such that the average wear of the segments is equal.
9. (withdrawn – currently amended) A method for serving concurrent search operations accessing a database file containing a number of documents, the method ~~containing~~ comprising the steps of:
 - incrementing, upon the launch of a new search operation to test a specific criterion on the documents in the database, an active search counter;
 - launching, in case the active search counter has a value indicating that one or more search operations are active and a document retrieve process is not yet running, the document retrieve process which consecutively and cyclically retrieves the documents contained in the database and provides them to any active search operations;
 - memorizing, in the launched new search operation receiving documents from the document retrieve process, as a first document an identifier of the document received first after launch, and applying the specific criterion on the received documents;
 - terminating, in case the search operation receives the first document a second time, the search operation and decrementing the active search counter;

- terminating, in case the active search counter has a value indicating that no more search operations are active, the document retrieve process.
10. (withdrawn) The method of claim 9, wherein the search operations originate from different sources.
11. (withdrawn) A data carrier having stored thereon a database file containing records wherein each record contains one distinct version number information monotonically related to the time when the record was written to the data carrier and stored with wrap-around in a fixed wordlength data field.
12. (withdrawn) The data carrier of claim 11, additionally wherein only that record is predefined to be considered valid, for which the subsequent record does not bear the subsequent version number.